

IN THE CLAIMS

1. (Currently Amended) A method for producing an output report comprising:

identifying a page template indicative of an output report having passive content, the passive content being static;

parsing tokens from the page template, the tokens being metadata tokens indicative of dynamic content, the dynamic content adapted to provide at least a portion of the output report, each of the metadata tokens corresponding to a javabean component such that parsing further comprises identifying the javabean component, the javabean component operable to process the metadata token;

retrieving, from a metadata repository, by the javabean component, metadata components corresponding to the metadata tokens and operable to provide the dynamic content corresponding to the parsed metadata tokens; and processing the metadata component by the javabean component;

for each token, generating the dynamic content, said generating further comprising fetching output data and processing display data, the output data operable to be retrieved from a dynamic repository and the display data operable to indicate organization of the fetched output data; generating the dynamic content further comprising:

retrieving, based on the metadata components, the display data and the output data from a repository, the display data and output data corresponding to the dynamic content for rendering on the output report;

processing the metadata components using the retrieved display data and output data to generate the dynamic content corresponding to the parsed metadata tokens; and

building the output report by assembling the generated dynamic content for each token in the page template, the page template indicative of a plurality of output reports, the dynamic content further comprising java server pages referencing metalanguage representations of at least portions of other java

server pages corresponding to the same rendered output report, building further comprising rendering an output page by:

fetching, based on the parsed token, a page descriptor corresponding to the parsed token, the page descriptor defining the metalanguage representation; retrieving the dynamic content from the fetched page descriptor; and
inserting the dynamic content in the output report by replacing the
metadata tokens.

2. (Previously Presented) The method of claim 1 further comprising receiving a user request for an HTML page, wherein said building further comprises rendering the HTML page responsive to the user request.

3. (Canceled)

4. (Previously Presented) The method of claim 1 wherein said building the output report further comprises

receiving an HTTP request on behalf of a user, the HTTP request indicative of an HTML response;

determining the page template corresponding to the requested HTML response;

generating, in a rendering engine, a JSP output page from the determined page template; and

transmitting the generated JSP output page to the user as an HTML page.

5. (Previously Presented) The method of claim 1 wherein the display data is formatted according to a predetermined syntax, the predetermined syntax operable for parsing and verifying the display data.

6. (Previously Presented) The method of claim 5 wherein the predetermined syntax is a different syntax than the page template and said generating further

comprises interpreting the display data from the predetermined syntax into the syntax defining the page template.

7. (Previously Presented) The method of claim 5 wherein the predetermined syntax includes syntactical elements, and is further operable for nesting the syntactical elements, the nested syntactical elements defining a hierarchical structure.

8. (Previously Presented) The method of claim 7 wherein the dynamic content is active content and the predetermined syntax conforms to XML, in which the syntactical elements further comprise XML tags.

9. (Original) The method of claim 8 wherein the predetermined syntax is a page descriptor syntax defined by an XML schema.

10. (Canceled)

11. (Original) The method of claim 10 wherein the metadata components further comprise page descriptors, the page descriptors conforming to a declarative syntax and indicative of the dynamic content.

12. (Canceled)

13. (Previously Presented) The method of claim 12 wherein the metadata component is an XML file including at least one page descriptor according to a predetermined syntax, and said processing further comprises retrieving the at least one page descriptors by the javabeen component and producing HTML code corresponding to the at least one page descriptors.

14. (Previously Presented) The method of claim 1 wherein the display data further comprises user specific views, the user specific views indicative of formatting and display preferences specific to a particular user and operable for interpretation by syntax processing components to render the display data according to the user specific views.

15. (Currently Amended) A data processing device responsive to an encoded set of processor based instructions, the instructions encoded in a memory of the data processing device for producing an output report comprising:

a web server operable to identify a page template indicative of an output report having passive content, the passive content being static the page template indicative of a plurality of output reports;

a parser operable to parse tokens from the page template, the tokens being metadata tokens indicative of dynamic content, the dynamic content adapted to provide at least a portion of the output report, the metadata token corresponding to a javabeen component and the parser further operable to identify the javabeen component, the javabeen component operable to process the metadata token, and the syntax processor is operable to invoke the javabeen component for retrieving and processing the metadata component;

a syntax processor operable to, for each token, generate the dynamic content, said generating further comprising fetching output data and processing display data, the output data operable to be fetched from a dynamic repository and the display data operable to indicate organization of the fetched output data, the syntax processor further operable to:

retrieve, based on the metadata components, the display data and the output data from a repository, the display data and output data corresponding to the dynamic content for rendering on the output report; and

process the metadata components using the retrieved display data and output data to generate the dynamic content

corresponding to the parsed metadata tokens, the syntax processor further operable to process the display data formatted according to a predetermined syntax, the predetermined syntax operable for parsing and verifying the display data,

the predetermined syntax being a different syntax than the page template and said generating further comprising interpreting the display data from the predetermined syntax into the syntax defining the page template,

wherein the predetermined syntax includes syntactical elements, and is further operable for nesting the syntactical elements, the nested syntactical elements defining a hierarchical structure; and

a formatter operable to build the output report by assembling the generated dynamic content for each token in the page template, the dynamic content further comprising java server pages referencing metalanguage representations of at least portions of other java server pages corresponding to the same rendered output report, building by the formatter further comprising rendering an output page by:

fetching, based on the parsed token, a page descriptor corresponding to the parsed token, the page descriptor defining the metalanguage representation;

retrieving the dynamic content from the fetched page descriptor; and
inserting the dynamic content in the output report by replacing the
metadata tokens.

16. (Previously Presented) The data processing device of claim 15 wherein the web server further comprises an interface operable to receive a user request for an HTML page, wherein said building further comprises rendering the HTML page responsive to the user request.

17. (Canceled)

-7-

18. (Previously Presented) The data processing device of claim 15 wherein the formatter is further operable to produce the output report by:

receiving an HTTP request on behalf of a user, the HTTP request indicative of an HTML response;

determining the page template corresponding to the requested HTML response;

generating, in a rendering engine, a JSP output page from the determined page template; and

transmitting the generated JSP output page to the user as an HTML page.

Claims 19-21. (Canceled)

22. (Original) The data processing device of claim 15 wherein the dynamic content is active content and the predetermined syntax conforms to XML, in which the syntactical elements further comprise XML tags.

23. (Original) The data processing device of claim 15 wherein the predetermined syntax is a page descriptor syntax defined by an XML schema.

24. (Canceled)

25. (Previously Presented) The data processing device of claim 24 wherein the metadata components further comprise page descriptors, the page descriptors conforming to a declarative syntax and indicative of the dynamic content.

26. (Canceled)

27. (Previously Presented) The data processing device of claim 26 wherein the metadata component is an XML file including at least one page descriptor according to the predetermined syntax, and the syntax processor is further

operable to retrieve the page descriptors via the javabeen components and produce HTML code corresponding to the at least one page descriptor.

28. (Previously Presented) The data processing device of claim 15 wherein the display data further comprises user specific views, the user specific views indicative of formatting and display preferences specific to a particular user and operable for interpretation by syntax processing components to render the display data according to the user specific views.

29. (Currently Amended) A computer program product having a computer readable medium storing computer program logic embodied in computer program code encoded thereon for producing an output report comprising:

computer program code for identifying a page template indicative of an output report having passive content, the passive content being static, the page template indicative of a plurality of output reports;

computer program code for parsing tokens from the page template, the tokens being metadata tokens indicative of dynamic content, the dynamic content adapted to provide at least a portion of the output report, each of the metadata tokens corresponding to a javabeen component such that parsing further comprises identifying the javabeen component, the javabeen component operable to process the metadata token;

retrieving, from a metadata repository, by the javabeen component, metadata components corresponding to the metadata tokens and operable to provide the dynamic content corresponding to the parsed metadata tokens; and processing the metadata component by the javabeen component;

computer program code operable for generating, for each parsed token, the dynamic content, said generating further comprising fetching output data and processing display data, the display data formatted according to a predetermined syntax, the predetermined syntax operable for parsing and verifying the display data, the output data operable to be retrieved from a

dynamic repository and the display data operable to indicate organization of the retrieved output data, the dynamic content further comprising java server pages referencing metalanguage representations of at least portions of other java server pages corresponding to the same rendered output report, generating the dynamic content further comprising:

retrieving, based on the metadata components, the display data and the output data from a repository, the display data and output data corresponding to the dynamic content for rendering on the output report;

processing the metadata components using the retrieved display data and output data to generate the dynamic content corresponding to the parsed metadata tokens; and

computer program code for building the output report by assembling the generated dynamic content for each token in the page template, building further comprising rendering an output page by:

fetching, based on the parsed token, a page descriptor corresponding to the parsed token, the page descriptor defining the metalanguage representation; and

retrieving the dynamic content from the fetched page descriptor, wherein the predetermined syntax comprises javabeen components operable to interpret the predetermined syntax into a syntax of the output page; and

inserting the dynamic content in the output report by replacing the metadata tokens;

computer program code for invoking the javabeen components for each of the metadata components to perform processing using the retrieved display data and the output data; and

computer program code for generating the dynamic content corresponding to each of the parsed metadata tokens.

30. (Currently Amended) An encoded set of processor based instructions having program code in an executable form on a computer readable storage medium for producing an output report comprising:

program code for identifying a page template indicative of an output report having passive content, the passive content being static, the page template indicative of a plurality of output reports;

program code for parsing tokens from the page template, the tokens being metadata tokens indicative of dynamic content, the dynamic content adapted to provide at least a portion of the output report, each of the metadata tokens corresponding to a javabeen component such that parsing further comprises:

identifying the javabeen component, the javabeen component operable to process the metadata token; and

retrieving, from a metadata repository, by the javabeen component, metadata components corresponding to the metadata tokens and operable to provide the dynamic content corresponding to the parsed metadata tokens;

program code for generating, for each token, the dynamic content, said generating further comprising fetching output data and processing display data, the output data operable to be fetched from a dynamic repository and the display data operable to indicate organization of the fetched output data, generating the dynamic content further comprising:

retrieving, based on the metadata components, the display data and the output data from a repository, the display data and output data corresponding to the dynamic content for rendering on the output report;

processing the metadata components using the retrieved display data and output data to generate the dynamic content corresponding to the parsed metadata tokens,

the display data formatted according to a predetermined syntax, the predetermined syntax operable for parsing and verifying the display data, the

predetermined syntax being a different syntax than the page template and said generating further comprising interpreting the display data from the predetermined syntax into the syntax defining the page template, the predetermined syntax including syntactical elements, and further operable for nesting the syntactical elements, the nested syntactical elements defining a hierarchical structure; and

program code for building the output report by assembling the generated dynamic content for each token in the page template, building further comprising rendering an output page by:

fetching, based on the parsed token, a page descriptor corresponding to the parsed token, the page descriptor defining the metalanguage representation; and
retrieving the dynamic content from the fetched page descriptor, the retrieved dynamic content further comprising java server pages referencing metalanguage representations of at least portions of other java server pages corresponding to the same rendered output report; and inserting the dynamic content in the output report by replacing the metadata tokens.

31. (Currently Amended) A processor based information processing device responsive to encoded set of processor based instructions encoded in a memory of the information processing device for producing an output report comprising:

means for identifying a page template indicative of an output report having passive content, the passive content being static, the page template indicative of a plurality of output reports;

means for parsing tokens from the page template, the tokens being metadata tokens indicative of dynamic content, the dynamic content adapted to provide at least a portion of the output report, each of the metadata tokens corresponding to a java bean component such that the means for parsing further comprises:

means for identifying the javabeen component, the javabeen component operable to process the metadata token; and

means for retrieving, from a metadata repository, by the javabeen component, metadata components corresponding to the metadata tokens and operable to provide the dynamic content corresponding to the parsed metadata tokens; and

processing the metadata component by the javabeen component;

means for generating, for each token, the dynamic content, said generating further comprising fetching output data and processing display data, the output data operable to be fetched from a dynamic repository and the display data operable to indicate organization of the fetched output data, generating the dynamic content further comprising:

retrieving, based on the metadata components, the display data and the output data from a repository, the display data and output data corresponding to the dynamic content for rendering on the output report; and

processing the metadata components using the retrieved display data and output data to generate the dynamic content corresponding to the parsed metadata tokens,

the display data formatted according to a predetermined syntax, the predetermined syntax operable for parsing and verifying the display data, the predetermined syntax being a different syntax than the page template and said generating further comprising interpreting the display data from the predetermined syntax into the syntax defining the page template, the predetermined syntax including syntactical elements, and further operable for nesting the syntactical elements, the nested syntactical elements defining a hierarchical structure; and

means for building the output report by assembling the generated dynamic content for each token in the page template, means for building further comprising means for rendering an output page by:

means for fetching, based on the parsed token, a page descriptor corresponding to the parsed token, the page descriptor defining the metalanguage representation;

means for retrieving the dynamic content from the fetched page descriptor
and

means for inserting the dynamic content in the output report by replacing the metadata tokens, the dynamic content further comprising java server pages referencing metalanguage representations of at least portions of other java server pages corresponding to the same rendered output report, the predetermined syntax comprising javabeen components operable to interpret the predetermined syntax into the syntax of the output page, wherein the means for generating further comprises:

means for invoking the javabeen components for each of the metadata tokens to perform processing using the display data and the fetched output data;
and

means for generating the dynamic content corresponding to each of the parsed tokens.

32. (Previously Presented) The data processing device of claim 15 wherein the predetermined syntax comprises javabeen components operable to interpret the predetermined syntax into a syntax of the output page, wherein the syntax processor:

invokes the javabeen components for each of the tokens to perform processing using the display data and the output data; and

generates the dynamic content corresponding to each of the parsed tokens.

33. (Previously Presented) The data processing device of claim 32 wherein the syntax processor is further operable to invoke the javabeen components to

interpret the predetermined XML compliant syntax of a metadata component corresponding to the tokens for display on a user device.

34. (Previously Presented) The data processing device of claim 1 wherein the predetermined syntax comprises javabeen components operable to interpret the predetermined syntax into a syntax of the output page, further comprising:

invoking the javabeen components for each of the tokens to perform processing using the display data and the output data; and

generating the dynamic content corresponding to each of the parsed tokens.

35. (Previously Presented) The data processing device of claim 34 wherein further comprising invoking the javabeen components to interpret the predetermined syntax, the syntax being XML compliant, of a metadata component corresponding to the token for display on a user device.

36. (Previously Presented) The method of claim 1 wherein a predetermined syntax comprises javabeen components operable to interpret the predetermined syntax into the syntax of the output page, further comprising

invoking the javabeen components for a metadata component[corresponding to the token to perform processing using the display data and the output data; and

generating the dynamic content corresponding to each of the parsed tokens.

37. (Previously Presented) The data processing device of claim 36 further comprising invoking the javabeen components to interpret the predetermined syntax, the predetermined syntax being XML compliant, of the metadata components for display on a user device.